

Proposed threshold-based and rule-based approaches to detecting duplicates in bibliographic database

M. Miftakul Amin^{1,2}, Deris Stiawan³, Ermatita³, Rahmat Budiarto⁴

¹Department of Computer Engineering, Politeknik Negeri Sriwijaya, Palembang, Indonesia

²Faculty of Engineering, Universitas Sriwijaya, Palembang, Indonesia

³Department of Computer Engineering, Faculty of Computer Science, Universitas Sriwijaya, Palembang, Indonesia

⁴Department of Computer Science, College of Computing and Information, Al-Baha University, Alaqiq, Saudi Arabia

Article Info

Article history:

Received Oct 4, 2023

Revised Feb 22, 2024

Accepted Feb 28, 2024

Keywords:

Duplicate detection

Research database

Rule-based

Similarity function

Threshold-based

ABSTRACT

Bibliographic databases are used to measure the performance of researchers, universities and research institutions. Thus, high data quality is required and data duplication is avoided. One of the weaknesses of the threshold-based approach in duplication detection is the low accuracy level. Therefore, another approach is required to improve duplication detection. This study proposes a method that combines threshold-based and rule-based approaches to perform duplication detection. These two approaches are implemented in the comparison stage. The cosine similarity function is used to create weight vectors from the features. Then, the comparison operator is used to determine whether the pair of records are grouped as duplication or not. Three research databases: Web of Science (WoS), Scopus, and Google Scholar (GS) on the Science and Technology Index (SINTA) database are investigated. Rule 4 and Rule 5 provide the best performance. For WoS dataset, the accuracy, precision, recall, and F1-measure values were 100.00%. For Scopus dataset, the accuracy and precision values were 100.00%, recall: 98.00%, and the F1-measure value is 98.00%. For GS dataset, the accuracy value was 100.00%, precision: 99.00%, recall: 97.00%, and the F1-measure value is 98.00%. The proposed method is potential tool for accurate detection on duplication records in publication databases.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Deris Stiawan

Department of Computer Engineering, Faculty of Computer Science, Universitas Sriwijaya

Palembang, Indonesia

Email: deris@unsri.ac.id

1. INTRODUCTION

Information is increasingly being stored electronically, it may be conveniently accessed and exchanged as both interaction and internet usage grow. Users have access to digital information sources at any time and from any location, and they can search for information collections based on their needs. These easily accessible electronic data collections can be used to disseminate knowledge in the field of research and education. Public understanding and scientific literacy can both rise with open and extensive access to scientific knowledge.

Amorim *et al.* [1] found that research on data governance is emerging as a major concern for researchers; therefore, universities and research institutes need to procure a number of tools that facilitate the management of scientific publication data. In the aspect of governance, meanwhile Heidorn [2] highlights that scientific asset management requires that information collections contain valid information. In the context of Indonesia, the *Garba Rujukan Digital* (GARUDA) database is one of several databases that play a

key role in integrating scientific publication data as research databases. According to Lukman *et al.* [3], the Science and Technology Index (SINTA) database acts as an indexer and grader for measuring research productivity. It aggregates data not only from research databases in Indonesia, but also from international scientific databases, such as the Web of Science (WoS), Google Scholar (GS), and Scopus.

Furthermore, in research carried out by Caragea *et al.* [4], the discovery of the same entity in several domains originating from different database sources is an important function in the digital world. The increase in information derived from heterogeneous data sources impacts the amount of information, which is always increasing, and issues with data quality. The large volume of data and the quality of the information it holds are two crucial attributes that have received significant attention from the perspective of information systems and databases [5]. One of the causes of low data quality is the duplication of records, causing the stored data to have low guaranteed validity [6]. Duplication detection is a difficult process, especially in databases with large volumes of data. As proposed by Trippel and Zinn [7], research databases should be collected, evaluated, and inventoried, and these databases should be accessible in a manner that is easy for those in need of the information. Research databases are compiled from various reliable and valid sources to provide quality research information resources.

Duplication detection has several equivalent terms, including entity resolution if detecting on one dataset and record linkage if detection involves external data sources [8]. Duplication detection addresses a particularly complex problem because the solution proffered must pay attention to aspects of the domain that need to be resolved, dataset characteristics (such as size and scheme), costs for processing training data, and evaluation of expected precision and recall [9].

Mishra *et al.* [10] have mentioned that one of the causes of duplication in a research database is that the researcher changes affiliation, such that some scientific publications produced by an author have different affiliation information. This matter can be reviewed in cases where an author is assigned a different affiliation at the time of publication [11]. Other factors that cause duplication are typography errors [12], omitted fields, and missing values [13]. The absence of a digital object identifier (DOI) in a scientific article also causes duplication [14]. According to research by Gyawali *et al.* [15], more than 82% of the papers collected in databases do not have a DOI.

Duplication is often discovered during the process of data integration [16]. This is because the integration process takes data from various sources. The same thing was said by Galhotra *et al.* [17], the main objective of the data integration process is to remove and detect duplication. The elimination of this duplication contributes to saving storage space and simplifying computation [18].

One important aspect of duplication detection is the determination of threshold values [19]. Selecting a high threshold value will result in a false negative, while choosing a low threshold value will result in a positive false value that cannot accurately detect duplicate information. The phenomenon of duplication in scientific publication databases has the negative impact of increasing scientific indicators without the addition of new knowledge.

Research conducted by Naumann and Herschel [20] shows a case, because of the large volume of research databases, these scientific databases are usually not integrated into a single database system but instead provide links to other representations in the database. Furthermore, Irawan *et al.* [21] report that in Indonesia, there are several major problems with sharing research databases, including particularly short data cycles and lack of initiative in certain aspects of managing research databases.

Mishra *et al.* [10] has developed a model called entity matching technique for bibliographic databases using two research databases: one from DBLP, and the other from ArnetMiner. This study uses cluster techniques and similarity functions to perform an entity matching process in the entity matching technique for bibliographic database model. The results of the measurements indicate a recall value of 88.23% and an F1-measure of 93.75%.

Locality-sensitive hashing and word embedding models have been implemented in the deduplication process in previous research [15]. The web API was developed to provide duplication detection services. The model in their study achieved the following results: a F1-score of 90.00% and an accuracy value of 90.30%. In other a model was developed for cleaning noisy metadata in research databases using a supervised machine learning approach [22]. The research databases investigated in this study are CiteSeerX, WoS, PubMed, and DBLP. The evaluation results include a precision value of 98.40% and an F1-measure value of 91.00%. The n-grams approach was adopted to perform matching on the WoS research databases proposed by [23]. The data collected include dates from 1991 to 2013. The similarity function was used to match the reference data in the research database, and the accuracy value obtained was 96.00%.

In a study performed by Fisher *et al.* [24], a data cleaning and matching model was developed using the Scopus research database. Based on the results of the first test using a threshold value of 1.00, a precision value of 85.90% was obtained. While using a threshold value of 0.90, a precision value of 87.00% was obtained. In the second test, using threshold values of 1.00 and 0.90, the same precision value of 96.00% was obtained for both threshold values.

Research conducted by Ektefa *et al.* [16] uses a threshold-based approach to detect duplication on a dataset comprising 864 records, including 112 duplicate records. The results show that the F-measure value was 99.10%. Furthermore, Ali *et al.* [25] detect duplication by considering the incomplete aspect of the data, which aims to improve accuracy so that clustering errors do not occur if there is incomplete information.

In their research, Jiang *et al.* [26] developed a metasearch engine that retrieves information from five research databases: PubMed, Embase, CINAHL, PsycINFO, and Cochrane Central Register of Controlled Trials. Their research produced seven rules used to detect duplication. However, they did not explain how to measure the performance of the developed model in their study. The information obtained using the model is the number of records identified as duplicates in the research database under investigation.

A supervised learning approach is adopted by Wu *et al.* [27] by implementing support vector machines, logistic regression, random forests, and naïve bayes models to perform entity matching on scholarly datasets. Their study reports an F1-measure rate of 90.00% using as many as 11 features. The study used two datasets: CiteSeerX and IEEE. Similar research has also been conducted by Sefid *et al.* [14] using four datasets: CiteSeerX, WoS, DBLP, and PubMed. A machine learning approach was adopted to perform entity matching using as many as seven features. The F1-measure yielded a value of 92.20%.

The rule-based approach is a trial-and-error-based method that requires human intervention in the form of adding and modifying rules to ensure that satisfactory results are obtained. Research performed by Paganelli *et al.* [28] developed rules in the Magellan ecosystem by developing a software library called TuneR, designed for ease of use, specifically for application developers. Rule formation considers three critical pieces of information: attributes, similarity functions, and threshold values. The results for testing on restaurant datasets using six rules were as: an average precision of 96.20%, recall of 89.45%, and an F1-measure value of 92.36%.

Research conducted by Ektefa *et al.* [16] and Fisher *et al.* [24] only use one method, namely threshold. So that in this study adding a rule-based approach to improve duplication detection results. Besides, some previous studies used datasets with small amounts, such as research conducted by Ektefa *et al.* [16] only used hundreds of records. So, this research will later present duplication detection on a large amount of data.

This study employs threshold-based and rule-based approaches to detect duplication in the SINTA database, which is used to score and measure the productivity of papers published by researchers and lecturers in Indonesia. Research data sources in the SINTA database were obtained from WoS, Scopus, and GS. These three research databases are investigated to detect data duplication. In this study, binary classification is performed to determine the status of record pairs, i.e., whether they were duplicates or not. This paper is organized as: section 2 describes the method. Section 3 describes the experimental results. Section 4 provides conclusions and future works.

2. METHOD

2.1. Dataset

This study uses datasets obtained from the SINTA database. In the SINTA database, there are three research databases obtained from WoS, Scopus, and GS. These three databases will be investigated for duplication. Table 1 presents information on the characteristics of the three databases. These three research database sources comprise of data on computer science, information systems, electrical engineering, and multimedia research fields.

Table 1. Record dataset SINTA

No.	Dataset	Number of records
1.	Author	15,059
2.	GS	450,679
3.	Scopus	75,467
4.	WoS	6,190

2.2. Proposed framework

This study follows the framework outlined in Figure 1. A research database is obtained from the SINTA database using several mechanisms, such as harvesting, metadata importing, crawling, and API consumption, from several heterogeneous research database sources [29]. After the research database is obtained, preprocessing is then performed, and continued with the stages of indexing, comparison, and classification. In the duplication detection process, the most important stages are candidate selection and candidate matching [30]. Duplication detection is used to produce a golden record in the research database, the one that is clean from data duplication.

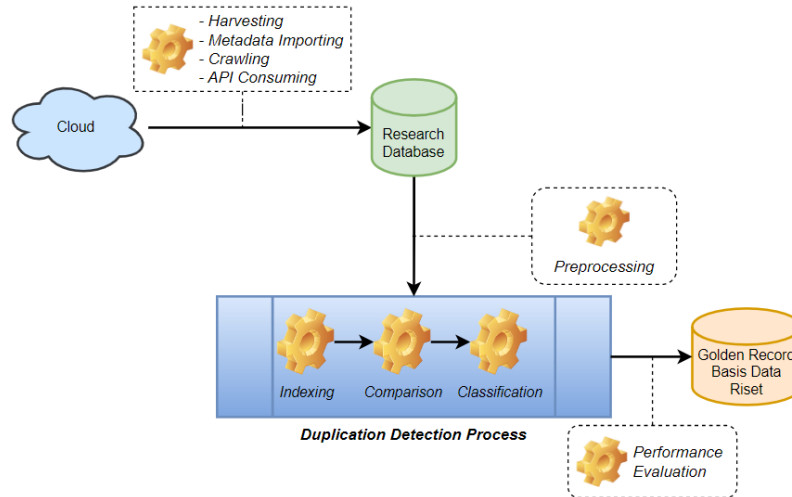


Figure 1. Proposed framework architecture

2.3. Similarity function

The duplicate detection process is done by comparing record pairs based on their degree of similarity for each attribute. The similarity function is used to measure the degree of similarity between the compared fields or records [31]. It plays a crucial role in duplication detection and acts as an algorithm when searching for data duplication [32]. In general, if the similarity function exceeds a certain threshold value, then the record pair will be classified into duplicate or not duplicate classes [33]. This similarity measurement produces a range of values falling between 0 and 1. This is the similarity value, which denotes the level of confidence in the similarity between two entities [34]. A value of 1 indicates that both items are exactly the same, while 0 indicates that the two items are different entities [35]. The similarity functions can be grouped as: i) character-based similarity, ii) numeric similarity, iii) token-based similarity, and iv) phonetic similarity [36]. The similarity value forms a vector to indicate whether two fields refer to the same entity. The cosine similarity function used in this study is formulated in (1). There are two n -dimensional vectors, V and W , and the cosine similarity function calculates the cosine value from the angle of α between these two vectors [20].

$$\text{CosineSimilarity}(V, W) = \cos(\alpha) = \frac{V \cdot W}{\|V\| \cdot \|W\|} \quad (1)$$

2.4. Threshold-based approach

The use of similarity functions requires a certain threshold value to perform duplication detection [37]. A threshold value is symbolised by a Θ sign, which means that the value is a determinant of decision [30] and a determinant of classification [38]. This threshold value is then combined with comparison operators, such as $<$, \leq , $>$, \geq , and $=$, to obtain the desired result [39].

2.5. Rule-based approach

The rule used in duplication detection is a rule for filtering a set of records [26], not a rule for performing the reasoning or inferencing process. The rule is built using several components, including attributes, the similarity function, operators, and threshold values [28], such that the rule is formed using the predicate $p: (a, f, op, thr)$, where $a \in A$, $f \in F$, $op \in O$, and $thr \in R$ are the threshold values. A dataset D comprises a set of records, which is d_1, \dots, d_N , while a record has an attribute value that can be denoted as $A = \{a_1, \dots, a_m\}$. The similarity function can be denoted in the form of $F = \{\text{edit}, \text{Jaccard}, \text{cosine}, \dots\}$, while the operator can be denoted in the form of comparison operators $O = \{>, =, <, \dots\}$. Thus, when there is a rule (name, edit, $>$, 0.6), it means that the record pair has a value of true if the edit distance function between the string names is more than 0.6.

Rule notation can also use the model described in [35]. For example, there is a rule that defines if the degree of similarity between the GivenName attribute between the r_i and r_j records is greater than 0.9, and the degree of similarity of the Surname attribute between the r_i record and the r_j record is equal to 1.0, then the r_i record and r_j record are grouped as the same record. The lowercase s symbol at the beginning of a field name denotes similarity. This model rule can be written as $(s(\text{GivenName})[r_i, r_j] > 0.9) \wedge (s(\text{Surname})[r_i, r_j] = 1.0) \Rightarrow [r_i, r_j] \rightarrow \text{Match}$.

2.6. Evaluation model

Once a model has been developed, it can be measured using a confusion matrix that compares actual and predicted classification results [40]. The confusion matrix comprises of true positive (TP), false positive (FP), false negative (FN), and true negative (TN) as classification evaluation parameters [41]. Accuracy is measurement of how close the truth of the results obtained is to the real results. In other words, accuracy is the comparison of the sum of all relevant results (the TPs and the TNs) with all results in the system, as expressed in (2). Precision is a measure commonly used in information retrieval to assess the quality of search results. Because precision does not include the number of TNs, it does not suffer from class imbalance problems, as opposed to accuracy; this is expressed in (3). Recall is the second most commonly used measure in information retrieval. Recall is similar to precision, as recall does not include the correct number of negatives (TNs). Recall does not suffer from class imbalance problems, as can be seen in (4). The F1-measure combines precision and recall and has a high value only if the precision and recall are high. Also known as F1-score, F1-measure, or F1-score, the F1-measure is used to calculate the harmonic mean between precision and recall, as expressed in (5).

$$accuracy = \frac{TP+TN}{TP+FP+TN+FN} \quad (2)$$

$$precision = \frac{TP}{TP+FP} \quad (3)$$

$$recall = \frac{TP}{TP+FN} \quad (4)$$

$$F1 - measure = 2 \times \left(\frac{prec \times rec}{prec+rec} \right) \quad (5)$$

3. RESULTS AND DISCUSSION

3.1. Indexing stage

The indexing process aims to determine the most optimal candidate pair; hence, not all candidate pairs will be processed to the next stage. The indexing stage reduces the number of FPs from the overall data processed. In a single database, the complete indexing process is implemented using (6), where A is the number of dataset records in a database [42]. Thus, the complete indexing of each dataset can be explained as (6):

$$|A| \times (|A|-1) / 2 \quad (6)$$

$$\begin{aligned} \text{WoS dataset} &: |A| \times (|A|-1) / 2 \\ &: |6,190| \times (|6,190|-1) / 2 \\ &: 19,154,955 \\ \text{Scopus dataset} &: |A| \times (|A|-1) / 2 \\ &: |75,467| \times (|75,467|-1) / 2 \\ &: 2,847,596,311 \\ \text{GS dataset} &: |A| \times (|A|-1) / 2 \\ &: |450,679| \times (|450,679|-1) / 2 \\ &: 101,555,555,181 \end{aligned}$$

Using a library record linkage toolkit, the distribution of several blocking mechanisms is presented in Table 2. Considering the data distribution, the field/record comparison stage uses the least number of candidate pairs, which are 806 in the WoS dataset, 65,541 in the Scopus dataset, and 904,008 in the GS dataset.

Table 2. Indexing process result on dataset

Parameter	WoS	Scopus	GS
Record total	6,190	75,467	450,679
Full indexing	19,154,955	2,847,596,311	101,555,555,181
Sorted neighbourhood (w=7)	21,918	447,476	2,907,615
Blocking (author_name)	81,117	1,908,836	29,293,573
Blocking (title)	806	65,541	904,056

This indexing process produces a distribution in the form of a distribution of pairwise record results with potential for duplication (Figure 2). The average of the generated indexing is no more than one percent of the entire record pair. This result shows that the indexing process of the duplication detection process is very effective at eliminating pairwise records that are indicated as not duplicate to be eliminated in the classification stage.

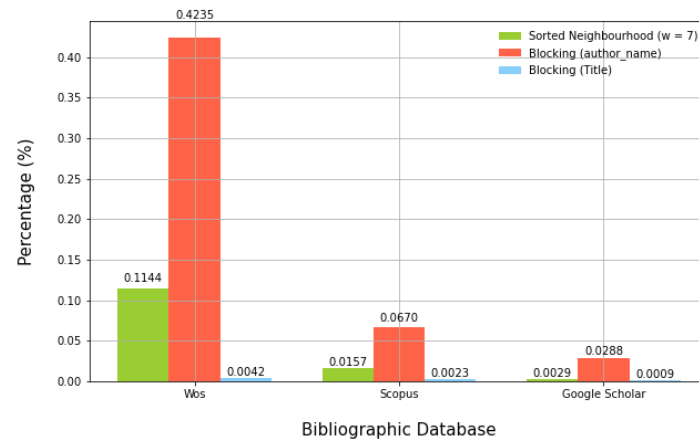


Figure 2. Distribution of indexing result

3.2. Comparison stage

After the candidate pair is obtained, the next stage is to compare the attributes used as determinants of whether there is a duplication case in the research database. There are four attributes used as determinants: author, title, venue (journal/proceedings), and the period represented by the year of publication. Each of these attributes is compared for degree of similarity and is then assigned a weight as the comparison result. In the next stage, each weight is accumulated into a weight vector. A model of this stage, as implemented in this study, is outlined in Figure 3, i.e., comparison of attributes in the dataset.

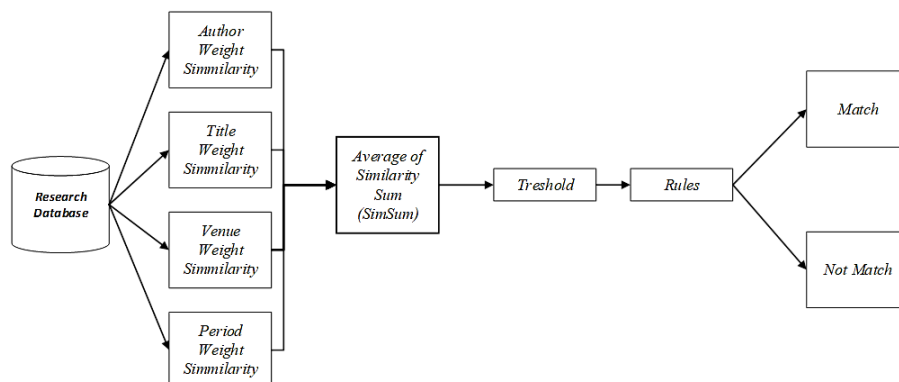


Figure 3. Model of field/record comparison calculation

Proposed threshold-based and rule-based approaches are used in the comparison stage. The rule generation is based on the threshold value obtained from the comparison of string similarity results derived from the four features mentioned. Then using the comparison operator will be determined, whether the pair of records are grouped as duplicate (match) or non-duplicate (not match).

3.3. Classification stage

In the classification stage, records are grouped into two classes: duplicate or non-duplicate. Notably, records with a threshold value (Θ) of 0.85 still have duplication; although each author has a different record, there is duplication because of the shared ownership among the authors of a scientific article. Figure 4 presents the visualisation result of a pairwise records distribution based on threshold values, where the

threshold value is higher proportional to the results of the number of record pairs getting smaller. Hence, it can be concluded that a large threshold value will precisely produce record pairs that are detected as duplicates. Figure 5 presents the results of a performance evaluation implemented with the threshold-based approach, and it can be seen that threshold values starting at 0.70 indicate increased accuracy, precision, and recall, indicating improvement in detection results.

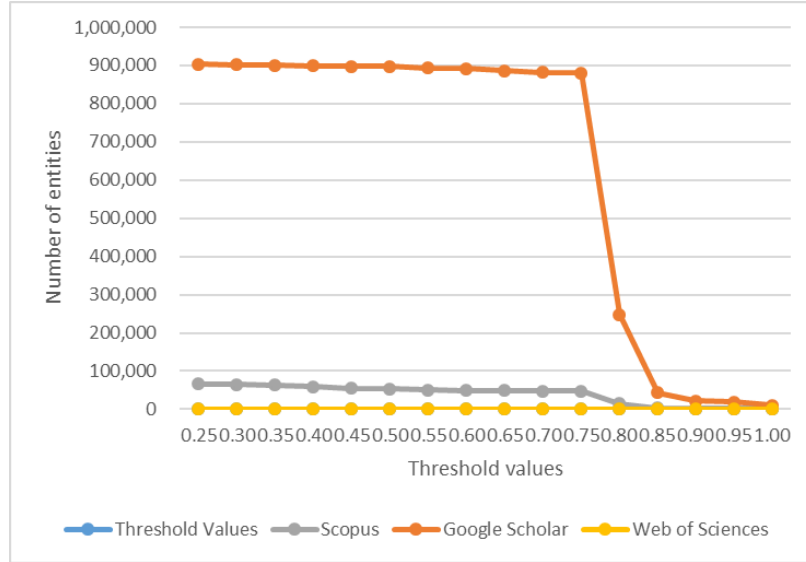


Figure 4. Graph of total pairwise record based on threshold values

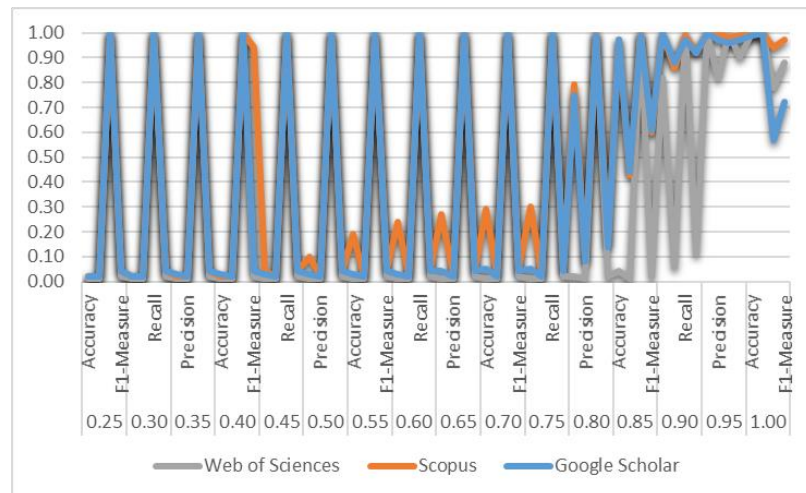


Figure 5. Graph of classification total based on threshold values

3.4. Performance improvement using rules

When threshold values are used at the classification stage, results are obtained that reflect the overall distribution. Rules are then added to filter the results based on several predetermined conditions. The value of 0.85 is based on the smallest distribution of value comparisons between fields in a ground-truth dataset created by experts. Some rules have been formulated with the following definitions:

Rule 1: $(s(author_name)[r_i, r_j] \geq 0.85) \wedge (s(title)[r_i, r_j] \geq 0.85) \Rightarrow [r_i, r_j] \rightarrow Match$

Rule 2: $(s(author_name)[r_i, r_j] \geq 0.85) \wedge (s(title)[r_i, r_j] \geq 0.85) \wedge (s(venue)[r_i, r_j] \geq 0.85) \Rightarrow [r_i, r_j] \rightarrow Match$

Rule 3: $(s(author_name)[r_i, r_j] \geq 0.85) \wedge (s(title)[r_i, r_j] \geq 0.85) \wedge (s(date_publish)[r_i, r_j] \geq 0.85) \Rightarrow [r_i, r_j] \rightarrow Match$

Rule 4: $(s(author_name)[r_i, r_j] \geq 0.85) \wedge (s(title)[r_i, r_j] \geq 0.85) \wedge (s(venue)[r_i, r_j] \geq 0.85) \wedge (s(date_publish)[r_i, r_j] \geq 0.85) \Rightarrow [r_i, r_j] \rightarrow Match$

Rule 5: $(s(author_name) [r_i, r_j] \geq 0.85) \wedge (s(title) [r_i, r_j] \geq 0.85) \wedge (s(venue) [r_i, r_j] \geq 0.85) \wedge (s(date_publish) [r_i, r_j] \geq 0.50) \Rightarrow [[r_i, r_j] \rightarrow Match$

3.5. Evaluation stage

After the model is obtained, the next stage involves measuring the level of accuracy of the model that has been developed. Based on the results in Figure 6, it appears that the threshold value is greater than 100.00% in all three datasets. In the WoS dataset, the recall value is 77.00%, and the F1-measure value is 88.00%. In the Scopus dataset, the recall value is 94.00%, and the F1-measure is 97.00%. In contrast, the GS dataset achieved an accuracy value of 99.00%, precision of 100.00%, recall of 57.00%, and F1-measure of 72.00%.

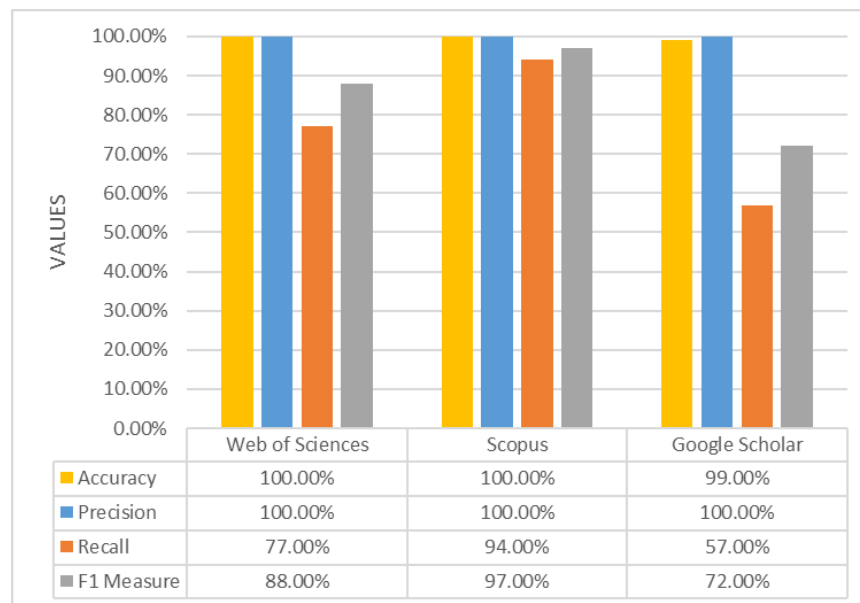


Figure 6. Threshold-based evaluation result with threshold value $\geq 100\%$

Table 3 presents the evaluation results of the rule-based approach. Based on the given threshold value of 100.00%, it impacts the duplication detection results, which still require improvements to the model's performance. It can be seen that some rules give varying results. So that it can be used as a reference in selecting rules in its application.

Table 3. Rule based performance method

Dataset	Measurement	Rule 1 (%)	Rule 2 (%)	Rule 3 (%)	Rule 4 (%)	Rule 5 (%)
WoS	Accuracy	99.00	100.00	100.00	100.00	100.00
	Precision	64.00	90.00	82.00	100.00	100.00
	Recall	100.00	100.00	100.00	100.00	100.00
	F1-measure	78.00	95.00	90.00	100.00	100.00
Scopus	Accuracy	99.00	100.00	100.00	100.00	100.00
	Precision	62.00	94.00	89.00	100.00	100.00
	Recall	99.00	98.00	97.00	96.00	98.00
	F1-measure	76.00	96.00	93.00	98.00	96.00
GS	Accuracy	100.00	100.00	100.00	100.00	100.00
	Precision	83.00	96.00	98.00	99.00	99.00
	Recall	100.00	98.00	97.00	96.00	97.00
	F1-measure	91.00	97.00	98.00	97.00	98.00

Figure 7 presents the results obtained with improved performance, using as many as five rules for duplication detection in WoS datasets. Rule 4 and Rule 5 provide the best detection results, with all measurement results achieving a value of 100.00%. The result indicates that both rules can be recommended for implementation.

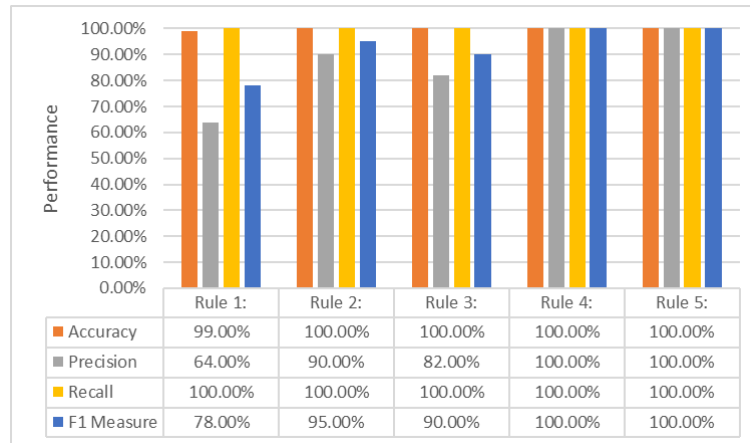


Figure 7. Rule-based evaluation result of WoS dataset

In the Scopus dataset, the results vary for each rule (Figure 8). It appears that Rule 4 and Rule 5 provide the best duplication detection results using four measurement parameters, with an average measurement result of 98.50%. Furthermore, for the GS dataset, the average results with four measurement parameters when using the different rules are as: 93.50% for Rule 1, Rule 2 of 97.75%, 98.25% for Rule 3, 98.00% for Rule 4, and 98.50% for Rule 5 (Figure 9). In this case, Rule 5 achieved the best measurement value.

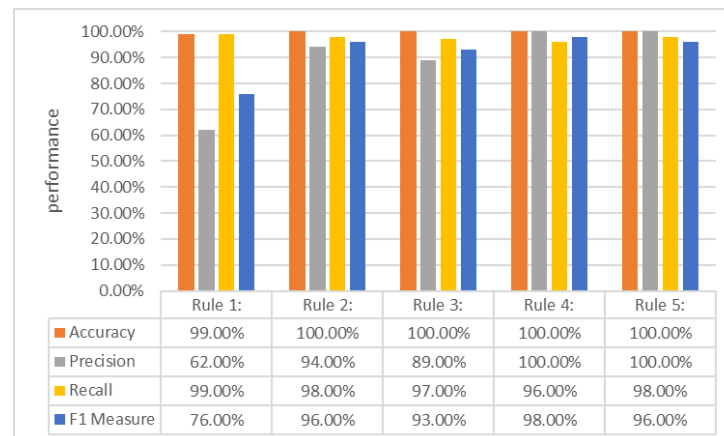


Figure 8. Rule-based evaluation result of Scopus dataset

In the aspect of model performance, our research produces better evaluation results than research by Caragea *et al.* [4] obtained an F1-measure value of 77.00% by implementing 3-grams on attribute titles using Jaccard similarity. Likewise, research conducted by Mishra *et al.* [10] using an attribute cluster approach to the duplication detection process, which obtained a recall value of 88.23% and an F1-measure of 93.75%. Where our research produces a recall value of 97.00%-100.00%, and an F1-measure value of 96.00%-100.00%. Likewise, compared to research conducted by Gyawali *et al.* [15] with the locality sensitivity hashing approach which obtained an F1-score value of 90.00% and an accuracy value of 90.30%.

In the computational aspect, our research uses a simpler calculation model compared to the research conducted by Sefid [22] who has developed a more complex supervised learning algorithm model. This research obtained a precision value of 98.40% and an F1-measure value of 91.00%.

From the three datasets that have been investigated in the duplication detection process, we found that the WoS dataset in terms of evaluation results has a measurement value of 100% for accuracy, precision, recall, and F1-measure values. This can be explained that in terms of data recording, WoS is quite neat with all fields filled in properly and completely. In contrast to the Scopus and GS datasets, there are still typographical errors, missing values, and not uniform values attached to each field. Referring to Figure 6 which only uses the threshold approach, the recall value and F1-measure obtained evaluation results in the range of 57.00% to 94.00% on the three datasets.

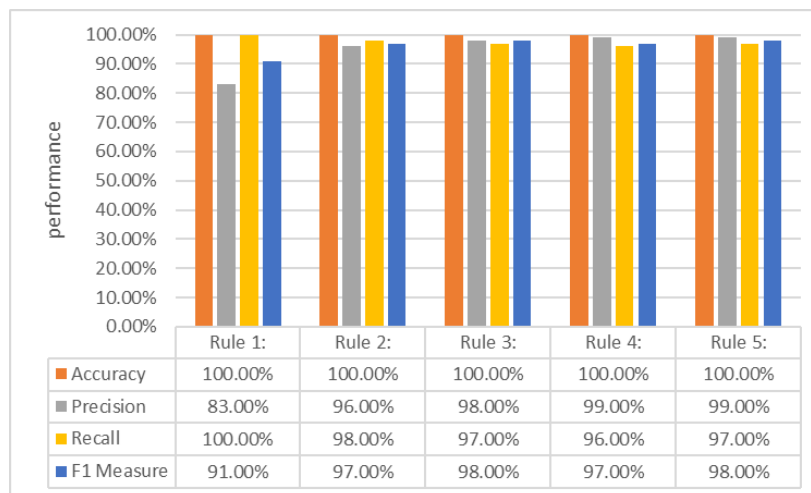


Figure 9. Rule-based evaluation result of GS dataset

4. CONCLUSION

A model for duplicate detection has been successfully created. The proposed method can be used as a reference for duplication detection, thereby reducing repeated recordings and improving data quality. Furthermore, this study has successfully combined the threshold-based and the rule-based approaches as a detection method on record duplication in research databases. The similarity value can be used as a reference for determining duplication detection in research databases. The incorporation of a rule-based approach in duplication detection improves the quality of the detection results compared to only using threshold-based approach. Based on the performance evaluation results, the proposed method showed that the optimal duplication detection results are obtained with Rule 4 and Rule 5. In the WoS dataset, the accuracy, precision, recall, and F1-measure values were 100.00% for both Rule 4 and Rule 5. In the Scopus dataset, the accuracy and precision values were 100.00% and 100.00%, recall values were 96.00% and 98.00%, and F1-measure values were 98.00% and 96.00% for Rule 4 and Rule 5, respectively. In the GS dataset, the accuracy values were 100.00% and 100.00%, the precision values were 99.00% and 99.00%, the recall values were 96.00% and 97.00%, and the F1-measure values were 97.00% and 98.00% for Rule 4 and Rule 5, respectively.

This study showed that duplication detection results depend on the quality of the similarity values between the fields being compared; hence, the duplication results are significantly influenced by the distribution of characters in each field. It needs to be investigated to conduct comparative studies of various similarity algorithms and methods to obtain the optimal similarity value. As a future work, the authors plan to investigate whether the proposed method can detect authorship, i.e., whether a scientific article is truly owned by the author whose information is stored in the dataset, as there are so many published studies in indexer databases such as GS whose ownership is not in accordance with a particular author, but is recorded as ownership of the author. Graph representation can be used as a model of the relationship between authors, which describes the ownership of a scientific article.

ACKNOWLEDGEMENTS

The authors would like to thank the leadership of the Universitas Sriwijaya and Politeknik Negeri Sriwijaya for supporting this research. This research is independent and funded by the researchers themselves and thanks to all parties.

REFERENCES




- [1] R. C. Amorim, J. A. Castro, J. Rocha da Silva, and C. Ribeiro, "A comparison of research data management platforms: architecture, flexible metadata and interoperability," *Universal Access in the Information Society*, vol. 16, no. 4, pp. 851–862, Nov. 2017, doi: 10.1007/s10209-016-0475-y.
- [2] P. B. Heidorn, "Shedding light on the dark data in the long tail of science," *Library Trends*, vol. 57, no. 2, pp. 280–299, 2008, doi: 10.1353/lib.0.0036.
- [3] L. Lukman *et al.*, "Proposal of the S-score for measuring the performance of researchers, institutions, and journals in Indonesia," *Science Editing*, vol. 5, no. 2, pp. 135–141, Aug. 2018, doi: 10.6087/kcse.138.
- [4] C. Caragea *et al.*, "CiteSeerx: A scholarly big dataset," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8416, pp. 311–322, 2014, doi: 10.1007/978-3-319-06028-6_26.
- [5] D. Hadzic and N. Sarajlic, "Methodology for fuzzy duplicate record identification based on the semantic-syntactic information of

- similarity,” *Journal of King Saud University - Computer and Information Sciences*, vol. 32, no. 1, pp. 126–136, Jan. 2020, doi: 10.1016/j.jksuci.2018.05.001.
- [6] G. Beskales, M. A. Soliman, I. F. Ilyas, and S. Ben-David, “Modeling and querying possible repairs in duplicate detection,” *Proceedings of the VLDB Endowment*, vol. 2, no. 1, pp. 598–609, Aug. 2009, doi: 10.14778/1687627.1687695.
 - [7] T. Trippel and C. Zinn, “Lessons learned: on the challenges of migrating a research data repository from a research institution to a university library,” *Language Resources and Evaluation*, vol. 55, no. 1, pp. 191–207, Mar. 2021, doi: 10.1007/s10579-019-09474-4.
 - [8] I. Koumarelas, L. Jiang, and F. Naumann, “Data Preparation for Duplicate Detection,” *Journal of Data and Information Quality*, vol. 12, no. 3, pp. 1–24, Sep. 2020, doi: 10.1145/3377878.
 - [9] F. Panse and F. Naumann, “Evaluation of duplicate detection algorithms: From quality measures to test data generation,” in *Proceedings - International Conference on Data Engineering*, IEEE, Apr. 2021, pp. 2373–2376, doi: 10.1109/ICDE51399.2021.00269.
 - [10] S. Mishra, S. Mondal, and S. Saha, “Entity matching technique for bibliographic database,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2013, pp. 34–41, doi: 10.1007/978-3-642-40173-2_5.
 - [11] S. Mishra, S. Saha, and S. Mondal, “An automatic framework for entity matching in bibliographic databases,” in *2016 IEEE Congress on Evolutionary Computation, CEC 2016*, IEEE, Jul. 2016, pp. 271–278, doi: 10.1109/CEC.2016.7743805.
 - [12] X. Chuan, W. Wei, L. Xuemin, and X. Y. Jeffrey, “Efficient similarity joins for near duplicate detection,” in *Proceeding of the 17th International Conference on World Wide Web 2008, WWW’08*, New York, NY, USA: ACM, Apr. 2008, pp. 131–140, doi: 10.1145/1367497.1367516.
 - [13] K. Deepa, R. Rangarajan, and M. S. Selvi, “Automatic Threshold Selection using PSO for GA based Duplicate Record Detection,” *International Journal of Computer Applications*, vol. 62, no. 4, pp. 22–27, Jan. 2013, doi: 10.5120/10068-4674.
 - [14] A. Sefid *et al.*, “Cleaning noisy and heterogeneous metadata for record linking across scholarly big datasets,” *Proceedings of the AAAI Conference on Artificial Intelligence*, Jul. 2019, vol. 33, no. 01, pp. 9601–9606, doi: 10.1609/aaai.v33i01.33019601.
 - [15] B. Gyawali, L. Anastasiou, and P. Knott, “Deduplication of scholarly documents using locality sensitive hashing and word embeddings,” *LREC 2020 - 12th International Conference on Language Resources and Evaluation, Conference Proceedings*, 2020, pp. 901–910.
 - [16] M. Ektefa, M. A. Jabar, F. Sidi, S. Memar, H. Ibrahim, and A. Ramli, “A threshold-based similarity measure for duplicate detection,” in *2011 IEEE Conference on Open Systems, ICOS 2011*, IEEE, Sep. 2011, pp. 37–41, doi: 10.1109/ICOS.2011.6079233.
 - [17] S. Galhotra, D. Firmani, B. Saha, and D. Srivastava, “BEER: Blocking for Effective Entity Resolution,” in *Proceedings of the ACM SIGMOD International Conference on Management of Data*, New York, NY, USA: ACM, Jun. 2021, pp. 2711–2715, doi: 10.1145/3448016.3452747.
 - [18] K. Williams and C. L. Giles, “Near duplicate detection in an academic digital library,” in *DocEng 2013 - Proceedings of the 2013 ACM Symposium on Document Engineering*, New York, NY, USA: ACM, Sep. 2013, pp. 91–94, doi: 10.1145/2494266.2494312.
 - [19] Y. V. Chekhovich and A. V. Khazov, “Analysis of duplicated publications in Russian journals,” *Journal of Informetrics*, vol. 16, no. 1, p. 101246, Feb. 2022, doi: 10.1016/j.joi.2021.101246.
 - [20] F. Naumann and M. Herschel, “An Introduction to Duplicate Detection,” *Synthesis Lectures on Data Management*, vol. 2, no. 1, pp. 1–87, Jan. 2010, doi: 10.2200/s00262ed1v01y201003dtm003.
 - [21] D. E. Irawan, C. Darujati, S. Soebandhi, F. Hayati, and D. A. P. Sari, “How to Extend your Data Lifetime: Research Data Management in Indonesia’s Context,” in *Proceedings of the 1st International Conference on Life, Innovation, Change and Knowledge (ICLICK 2018)*, Paris, France: Atlantis Press, 2019, doi: 10.2991/iclick-18.2019.33.
 - [22] A. Sefid, “Record Linkage Between CiteSeerX and Scholarly Big Datasets,” M.S. thesis, Computer Science and Engineering, The Pennsylvania State University, Pennsylvania, United States, 2019.
 - [23] M. A. Abdulhayoglu, B. Thijs, and W. Jeuris, “Using character n-grams to match a list of publications to references in bibliographic databases,” *Scientometrics*, vol. 109, no. 3, pp. 1525–1546, Dec. 2016, doi: 10.1007/s11192-016-2066-3.
 - [24] J. Fisher, Q. Wang, P. Wong, and P. Christen, “Data cleaning and matching of institutions in bibliographic databases,” *Conferences in Research and Practice in Information Technology Series*, vol. 146, pp. 139–148, 2013.
 - [25] A. Ali, N. A. Emran, and S. A. Asmai, “Missing values compensation in duplicates detection using hot deck method,” *Journal of Big Data*, vol. 8, no. 1, p. 112, Dec. 2021, doi: 10.1186/s40537-021-00502-1.
 - [26] Y. Jiang, C. Lin, W. Meng, C. Yu, A. M. Cohen, and N. R. Smalheiser, “Rule-based deduplication of article records from bibliographic databases,” *Database*, vol. 2014, Jan. 2014, doi: 10.1093/database/bat086.
 - [27] J. Wu, A. Sefid, A. C. Ge, and C. L. Giles, “A supervised learning approach to entity matching between scholarly big datasets,” in *Proceedings of the Knowledge Capture Conference, K-CAP 2017*, New York, NY, USA: ACM, Dec. 2017, pp. 1–4, doi: 10.1145/3148011.3154470.
 - [28] M. Paganelli, F. Guerra, P. Sottovia, and Y. Velegrakis, “TuneR: Fine tuning of rule-based entity matchers,” in *International Conference on Information and Knowledge Management, Proceedings*, New York, NY, USA: ACM, Nov. 2019, pp. 2945–2948, doi: 10.1145/3357384.3357854.
 - [29] I. M. I. Subroto, T. Sutikno, and D. Stiawan, “The architecture of indonesian publication index: A major indonesian academic database,” *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 12, no. 1, pp. 1–5, Mar. 2014, doi: 10.12928/TELKOMNIKA.v12i1.1790.
 - [30] G. Papadakis, D. Skoutas, E. Thanos, and T. Palpanas, “Blocking and Filtering Techniques for Entity Resolution: A Survey,” *ACM Computing Surveys*, vol. 53, no. 2, pp. 1–42, Mar. 2020, doi: 10.1145/3377455.
 - [31] V. Wandhekar and A. Mohanpurkar, “Validation of Deduplication in Data using Similarity Measure,” *International Journal of Computer Applications*, vol. 116, no. 21, pp. 18–22, Apr. 2015, doi: 10.5120/20460-2819.
 - [32] D. Hadzic, N. Sarajlic, and J. Malkic, “Different similarity measures to identify duplicate records in relational databases,” in *24th Telecommunications Forum, TELFOR 2016*, IEEE, Nov. 2017, pp. 1–4, doi: 10.1109/TELFOR.2016.7818899.
 - [33] L. Leitão, P. Calado, and M. Herschel, “Efficient and effective duplicate detection in hierarchical data,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 5, pp. 1028–1041, May 2013, doi: 10.1109/TKDE.2012.60.
 - [34] S. R. Yerva, Z. Miklós, and K. Aberer, “Quality-aware similarity assessment for entity matching in Web data,” *Information Systems*, vol. 37, no. 4, pp. 336–351, Jun. 2012, doi: 10.1016/j.is.2011.09.007.
 - [35] P. Christen, *Data matching: Concepts and techniques for record linkage, entity resolution, and duplicate detection*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, doi: 10.1007/978-3-642-31164-2.
 - [36] D. Bharambe, S. Jain, and A. Jain, “A Survey : Detection of Duplicate Record,” *International Journal of Emerging Technology and Advanced Engineering*, vol. 2, no. 11, pp. 298–307, 2012.




- [37] J. B. Dos Santos, C. A. Heuser, V. P. Moreira, and L. K. Wives, "Automatic threshold estimation for data matching applications," *Information Sciences*, vol. 181, no. 13, pp. 2685–2699, Jul. 2011, doi: 10.1016/j.ins.2010.05.029.
- [38] E. Joffe *et al.*, "A benchmark comparison of deterministic and probabilistic methods for defining manual review datasets in duplicate records reconciliation," *Journal of the American Medical Informatics Association*, vol. 21, no. 1, pp. 97–104, Jan. 2014, doi: 10.1136/amiajnl-2013-001744.
- [39] J. Wang, G. Li, J. X. Yu, and J. Feng, "Entity matching: How similar is similar," *Proceedings of the VLDB Endowment*, vol. 4, no. 10, pp. 622–633, Jul. 2011, doi: 10.14778/2021017.2021020.
- [40] S. Maity, N. Das, M. Majumder, and D. R. Dasadhikary, "Word Embedding and String-Matching Techniques for Automobile Entity Name Identification from Web Reviews," *EAI Endorsed Transactions on Scalable Information Systems*, vol. 8, no. 33, p. 169918, Oct. 2021, doi: 10.4108/EAI14-5-2021.169918.
- [41] J. Website, "International Journal of Evaluation measure for group-based record linkage," *International Journal of Population Data Science*, vol. 4, no. 1, 2019.
- [42] P. Christen, "A survey of indexing techniques for scalable record linkage and deduplication," *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 9, pp. 1537–1555, Sep. 2012, doi: 10.1109/TKDE.2011.127.

BIOGRAPHIES OF AUTHORS






M. Miftakul Amin    received the M.Eng. degree in Computer and Information System from Universitas Gadjah Mada, Indonesia. He is currently a Senior Lecturer with the Department of Computer Engineering, Politeknik Negeri Sriwijaya, Indonesia. His research interests include software engineering, decision support system, data mining, and machine learning. He can be contacted at email: miftakul_a@polsri.ac.id.






Deris Stiawan    received his Ph.D. degree in Computer Engineering from Universiti Teknologi Malaysia, Malaysia. He is currently a Professor with the Faculty of Computer Science, Universitas Sriwijaya. His research interests include computer networks, intrusion detection/prevention systems, and heterogeneous networks. He can be contacted at email: deris@unsri.ac.id.



Ermataita    received the Ph.D. degree in Computer Science from Universitas Gadjah Mada, Indonesia. He is currently a Senior Lecturer with the Department of Computer Engineering, Faculty of Computer Science, Universitas Sriwijaya. Her research interests include computer science, decision support system, data mining, machine learning, and education technology. She can be contacted at email: ermatita@ilkom.unsri.ac.id.



Rahmat Budiarto    received the B.Sc. degree from Bandung Institute of Technology, in 1986, the M.Eng. and Dr.Eng. degrees in computer science from the Nagoya Institute of Technology, in 1995 and 1998, respectively. He is currently a Full Professor with the College of Computer Science and IT, Albaha University, Saudi Arabia. His research interests include intelligent systems, brain modeling, IPv6, network security, wireless sensor networks, and MANETs. He can be contacted at email: rahmat@bu.edu.sa.